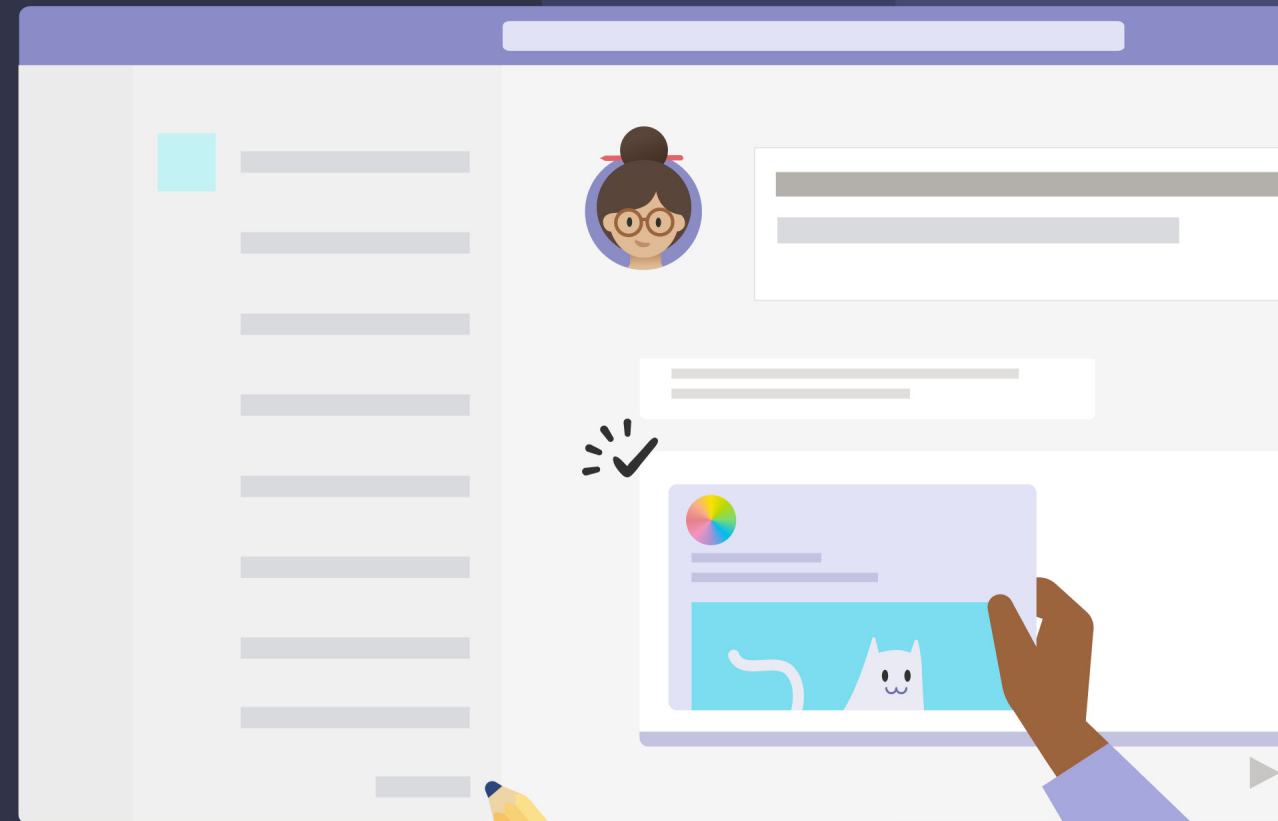




Best Practices

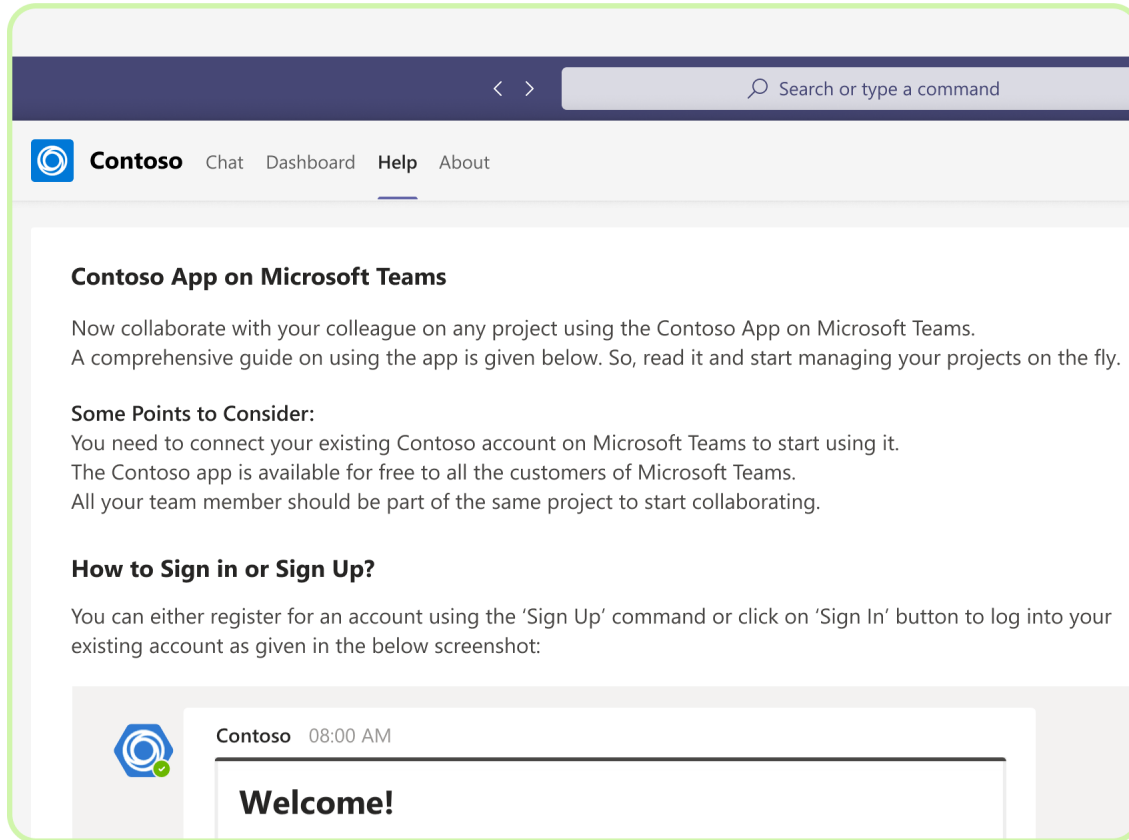
Engaging customers on the Teams app



Tip: 1

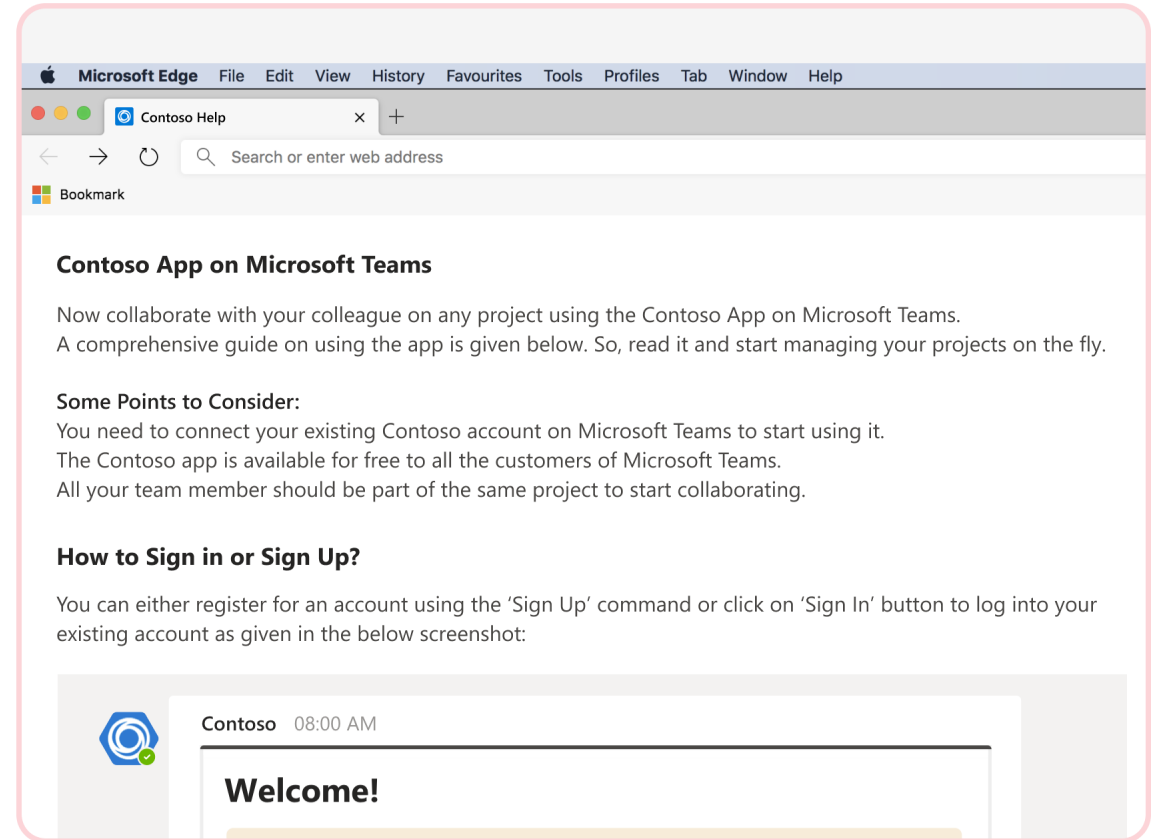
Engage customers within Teams

Engage customers



✔ Do: Use task modules or tabs

All core and non-core scenarios should be completed end-to-end inside the Teams experience. For instance, if a user clicks on a button or link for help, deep-link them to the help tab instead of opening a browser window of the help page on your website.



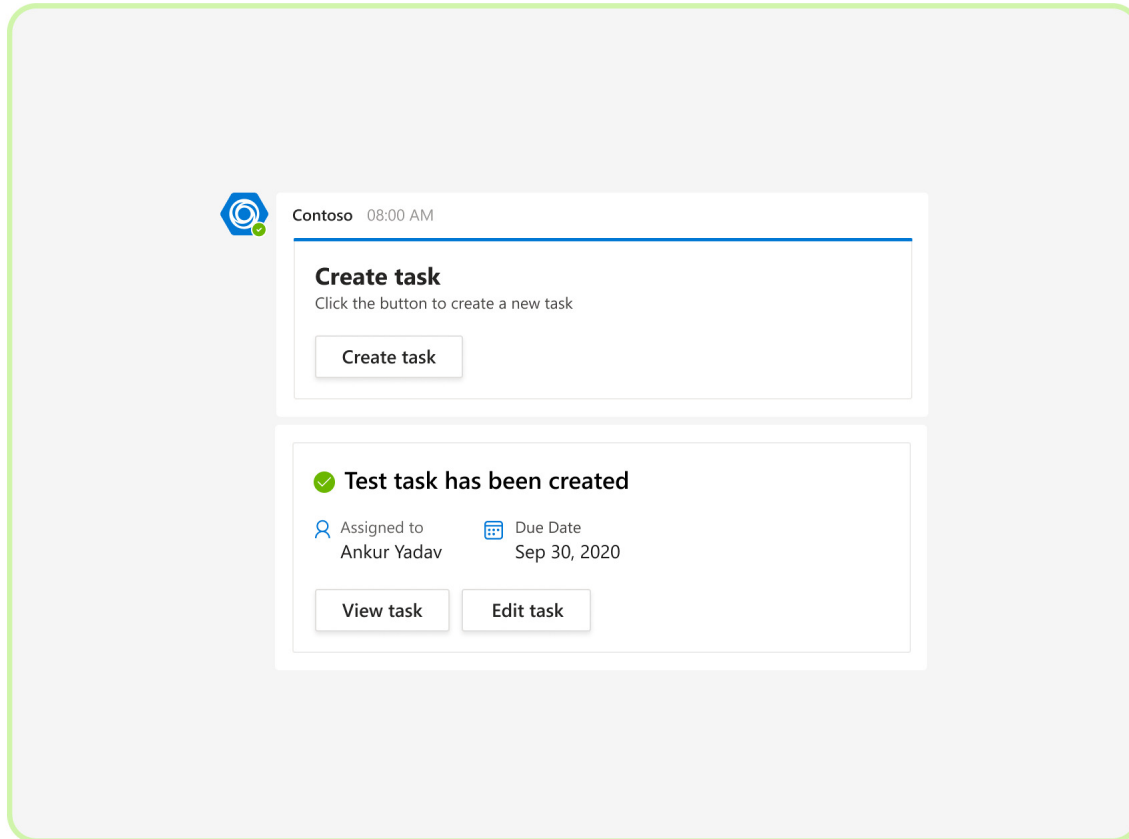
✘ Don't: Take users away from the conversation

Don't direct users to a web page outside Teams.

Tip: 2

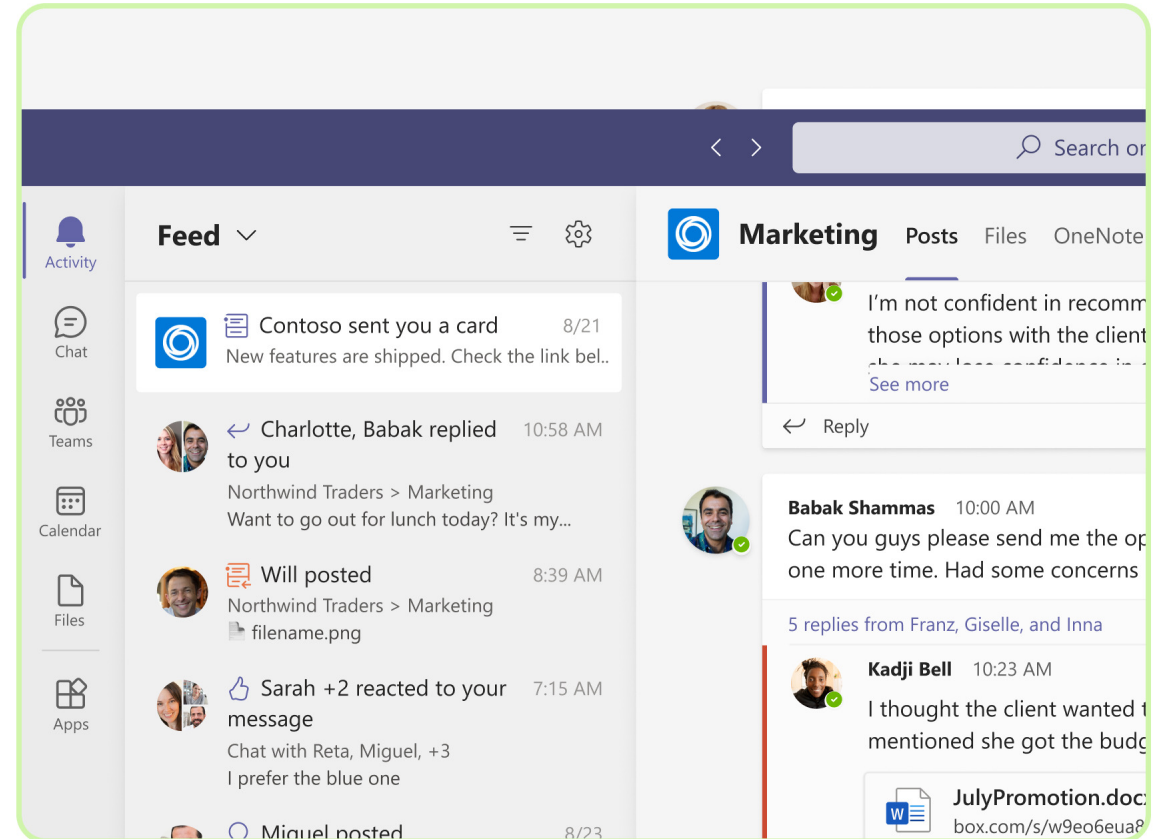
Always notify updates

New updates



✔ Do: Always notify any relevant updates

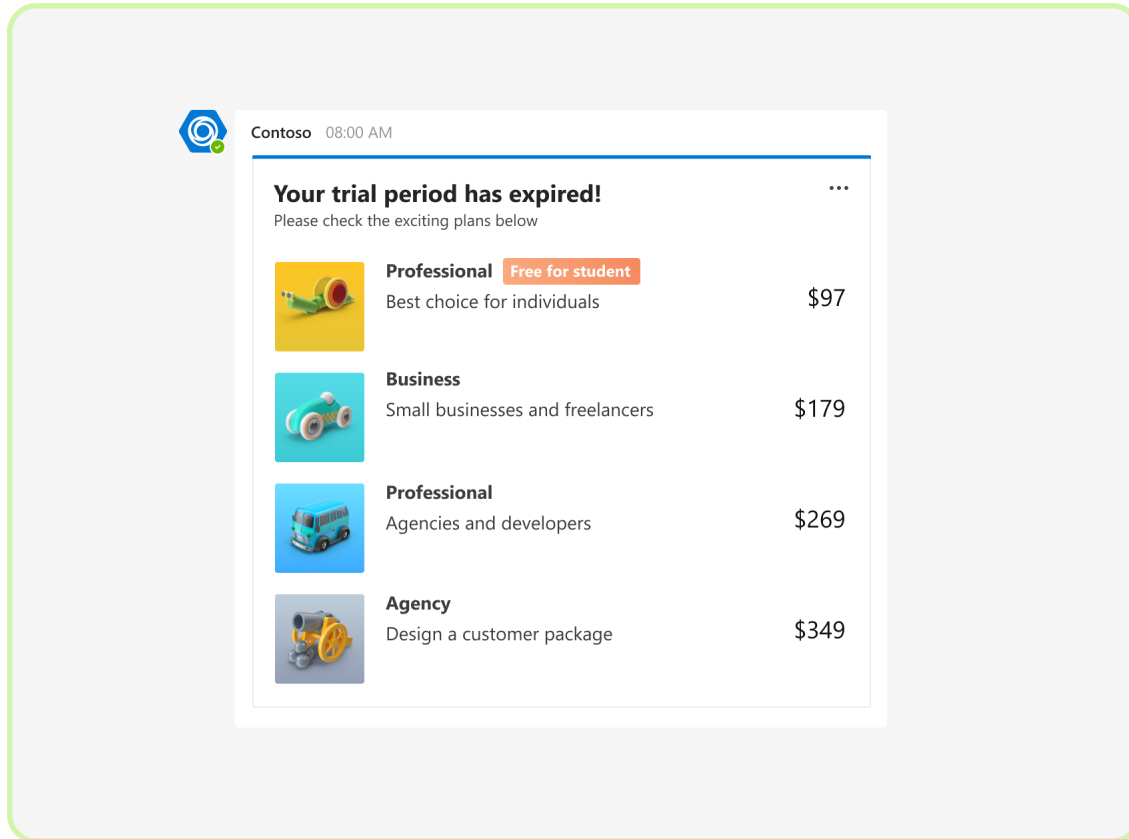
Example 1: If there are any status changes in the app, do send a bot message to let the users know about it.



✔ Do: Always notify any relevant updates

Example 2: If you update your app, do inform your users about new features that have gone live in the form of a notification.

New updates



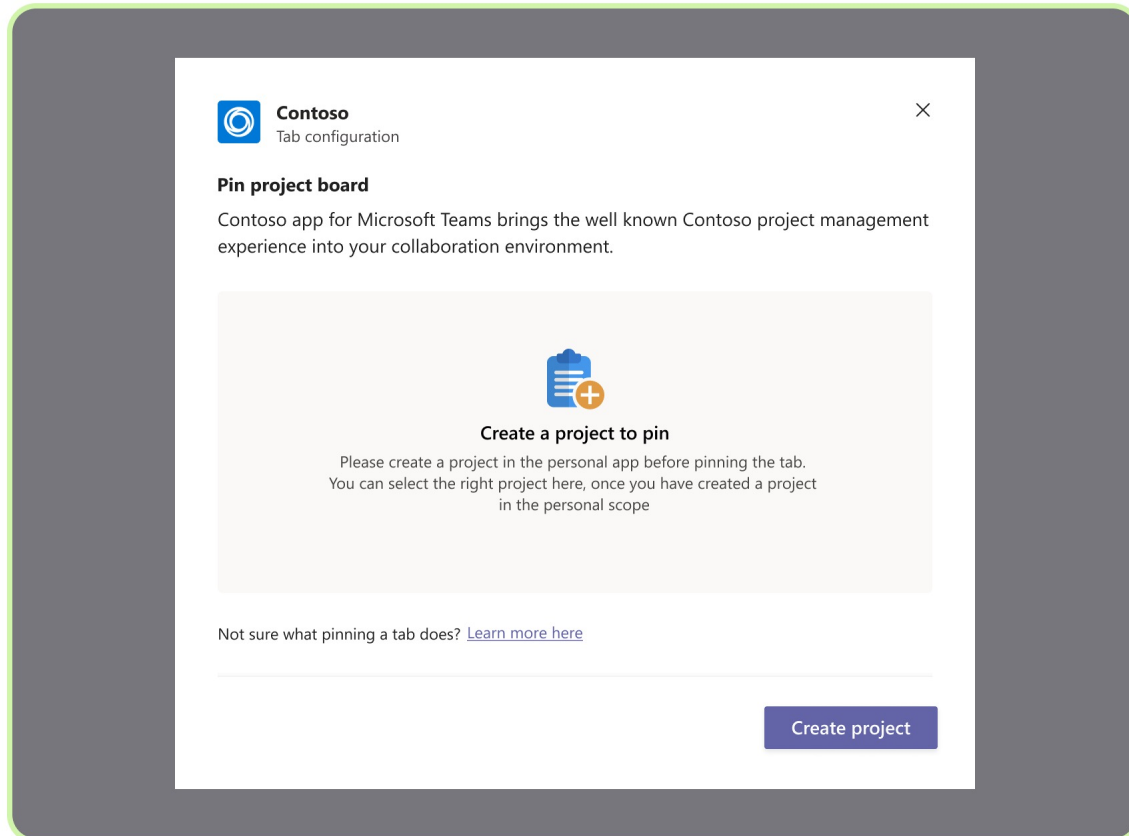
✔ Do: Always notify any relevant updates

Example 3: In case if you have a freemium app, do inform your users about the pricing or subscription plans once the trial period is over.

Tip: 3

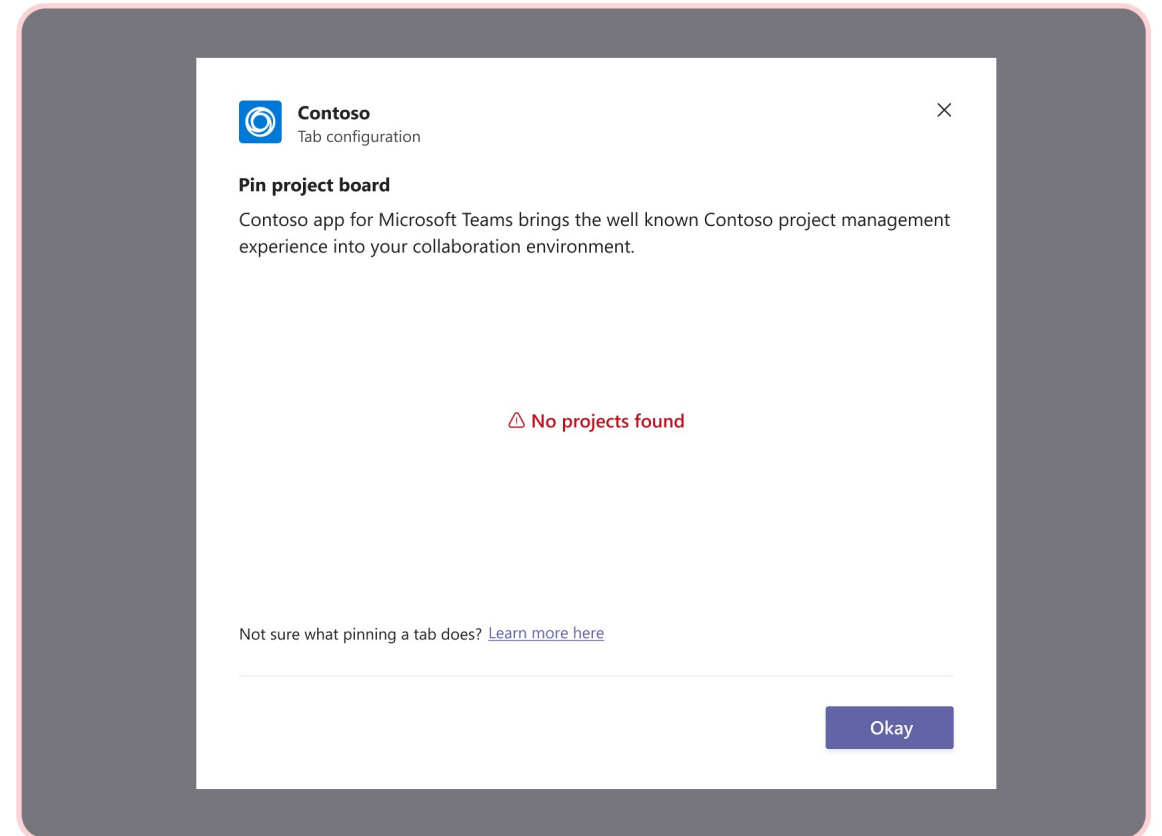
Clear next steps on Tab Configuration page

Tab configuration



✔ Do: Clear next steps on Tab Configuration page

If the tab configuration page requires a few steps to be done before the tab can be pinned, elucidate them clearly to the users. For instance, if a project must be created in a personal app before it can be pinned in collaborative scope, show a button to the users so that they can easily go to the bot/tab to create a board.



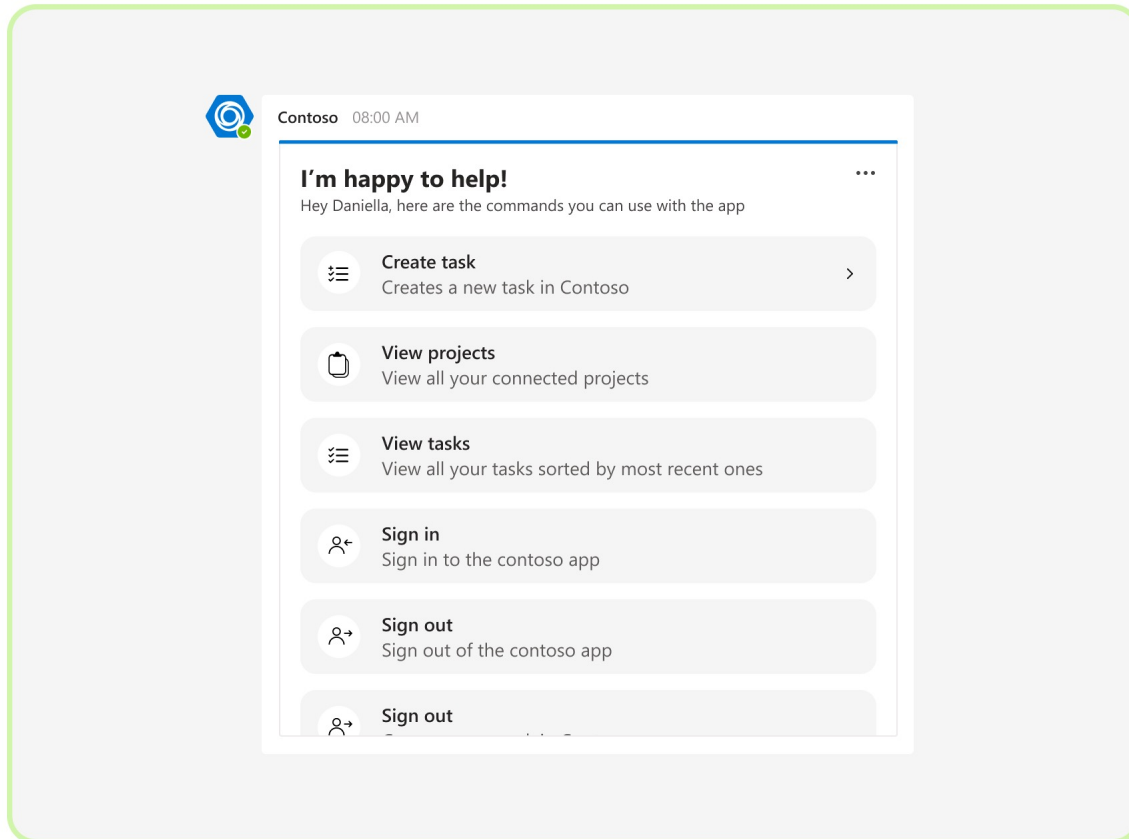
✘ Don't: Show dead ends to the users on Tab Configuration page

Avoid just showing the errors on the tab configuration page without stating the reason for the error. Don't forget to provide a CTA for the next step to resolve the problem.

Tip: 4

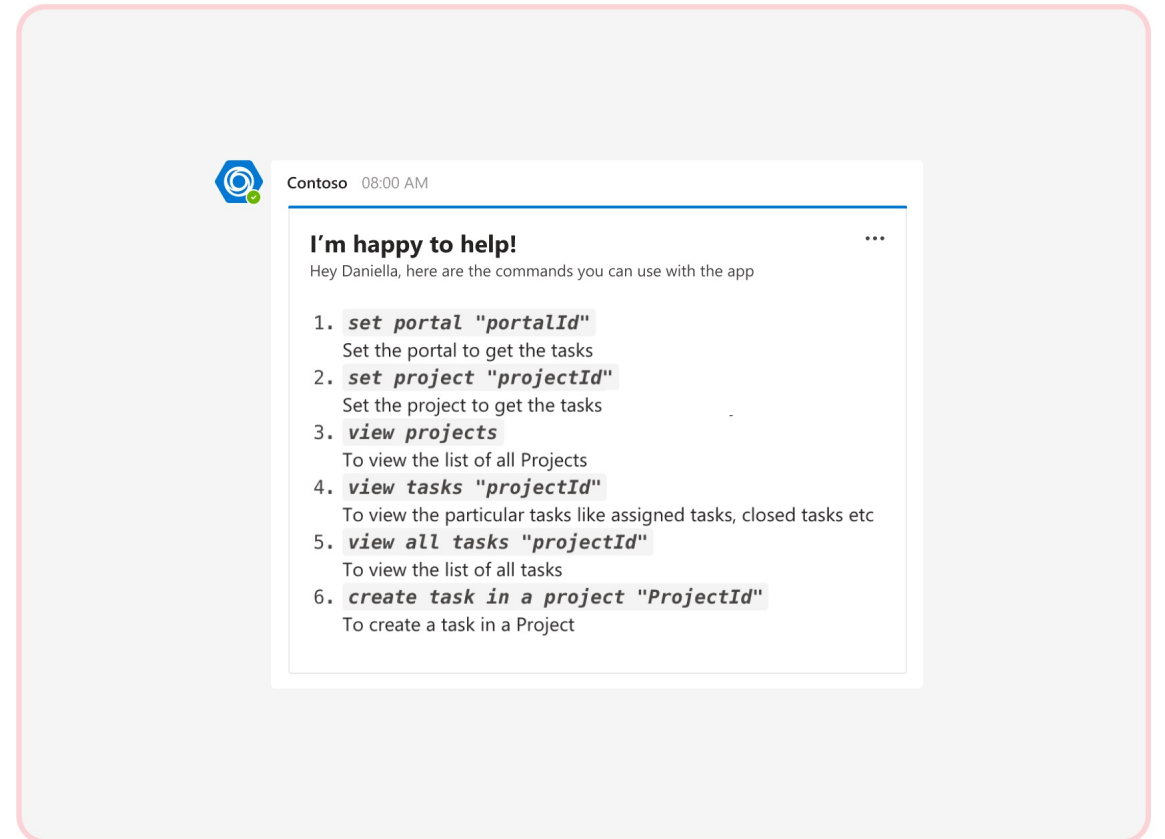
Use adaptive cards for Help command

Help card



✔ Do: Use Adaptive Cards for Help command

It is recommended to use an adaptive card to display Help cards. You can either use buttons (mapping to each of the supported command) or a list card to implement this. If there are more than 6 commands, use multi-level buttons or the last button can be titled as 'More' to display other commands in the next adaptive card.



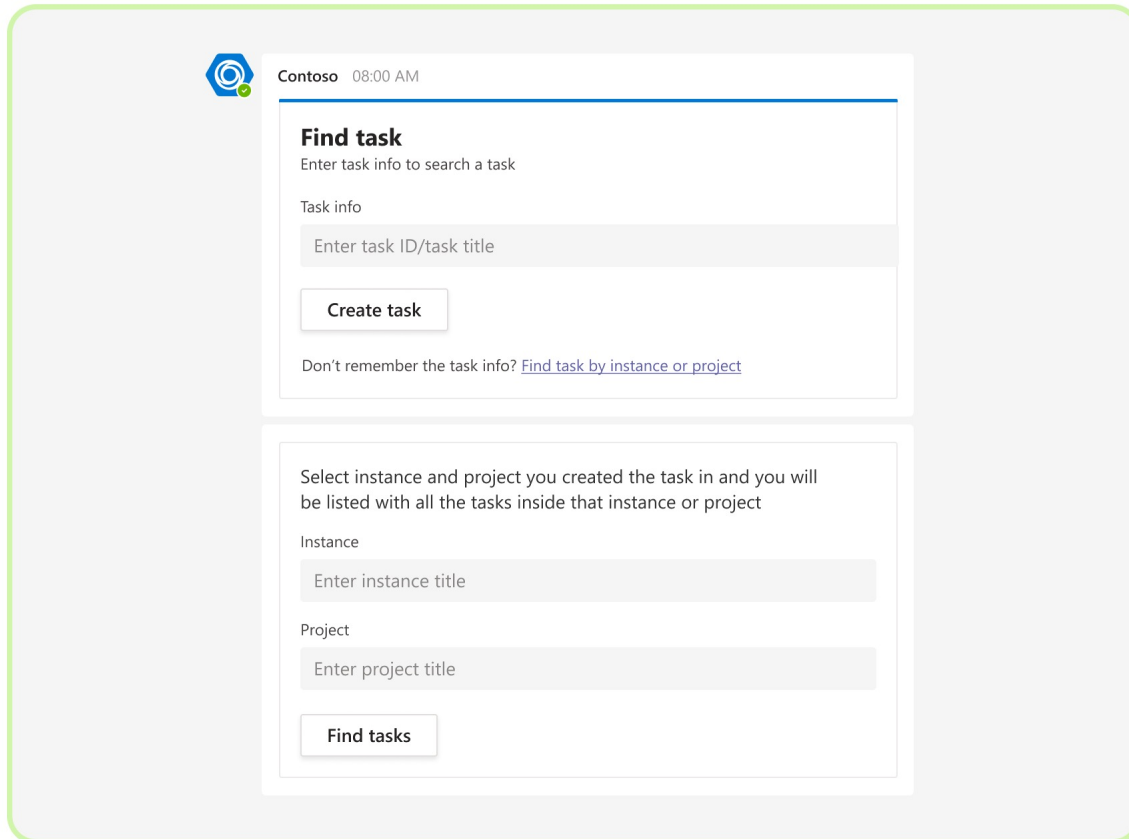
✘ Don't: Use just textual content for Help command

Avoid showing just the text as a response to help command. Textual content are difficult to read and inconvenient for the users to type the commands instead of just clicking on buttons or list items.

Tip: 5

Type less, Click more

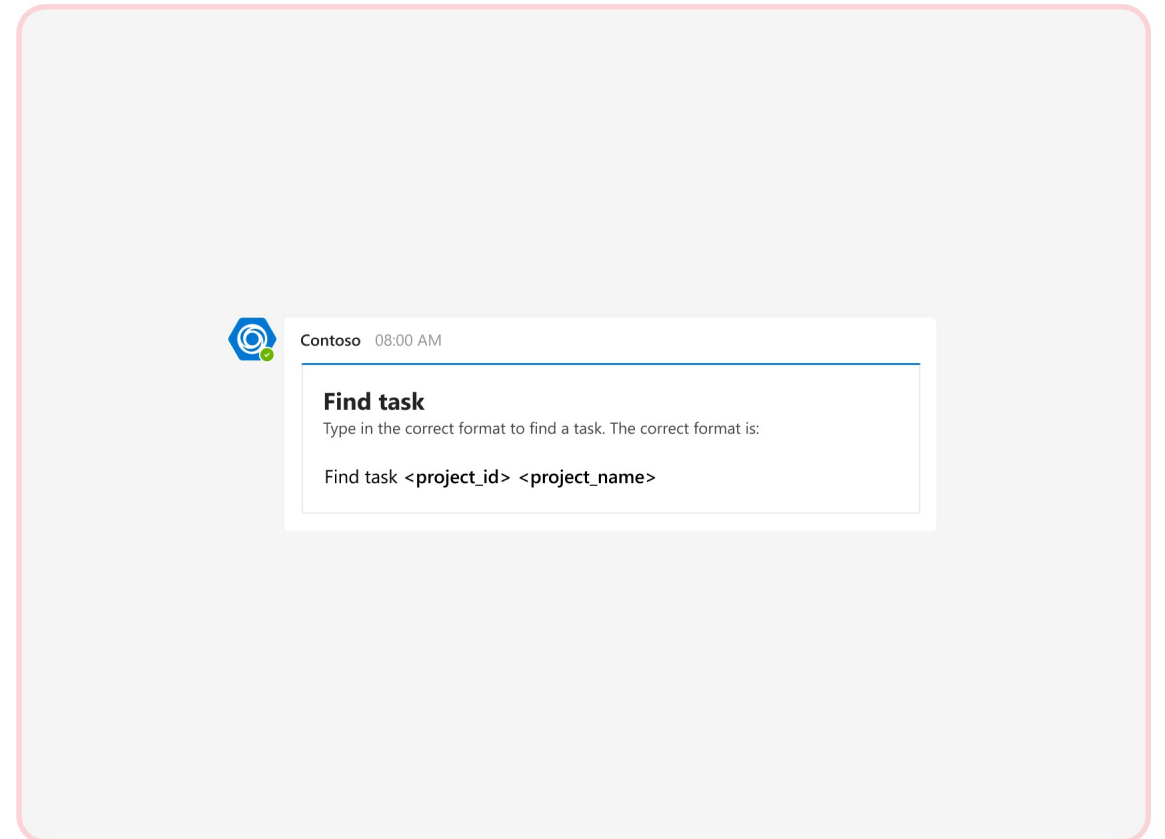
Type less, Click more



The screenshot shows a Teams chat window from 'Contoso' at 08:00 AM. The chat contains a card titled 'Find task' with the instruction 'Enter task info to search a task'. Below this is a 'Task info' section with a text input field labeled 'Enter task ID/task title' and a 'Create task' button. A link 'Find task by instance or project' is provided. Below the card is another section with the instruction 'Select instance and project you created the task in and you will be listed with all the tasks inside that instance or project'. It contains two text input fields: 'Instance' (labeled 'Enter instance title') and 'Project' (labeled 'Enter project title'), followed by a 'Find tasks' button.

✔ Do: Use Fluent UI components for user input

If an input must be taken from the user, then use input field, dropdown, radio, button and other Teams Fluent UI components for the standard or template text. Users should be asked to type only non-standard or custom text.



The screenshot shows a Teams chat window from 'Contoso' at 08:00 AM. The chat contains a card titled 'Find task' with the instruction 'Type in the correct format to find a task. The correct format is:'. Below this is a text input field containing the command 'Find task <project_id> <project_name>'. The card is highlighted with a red border.

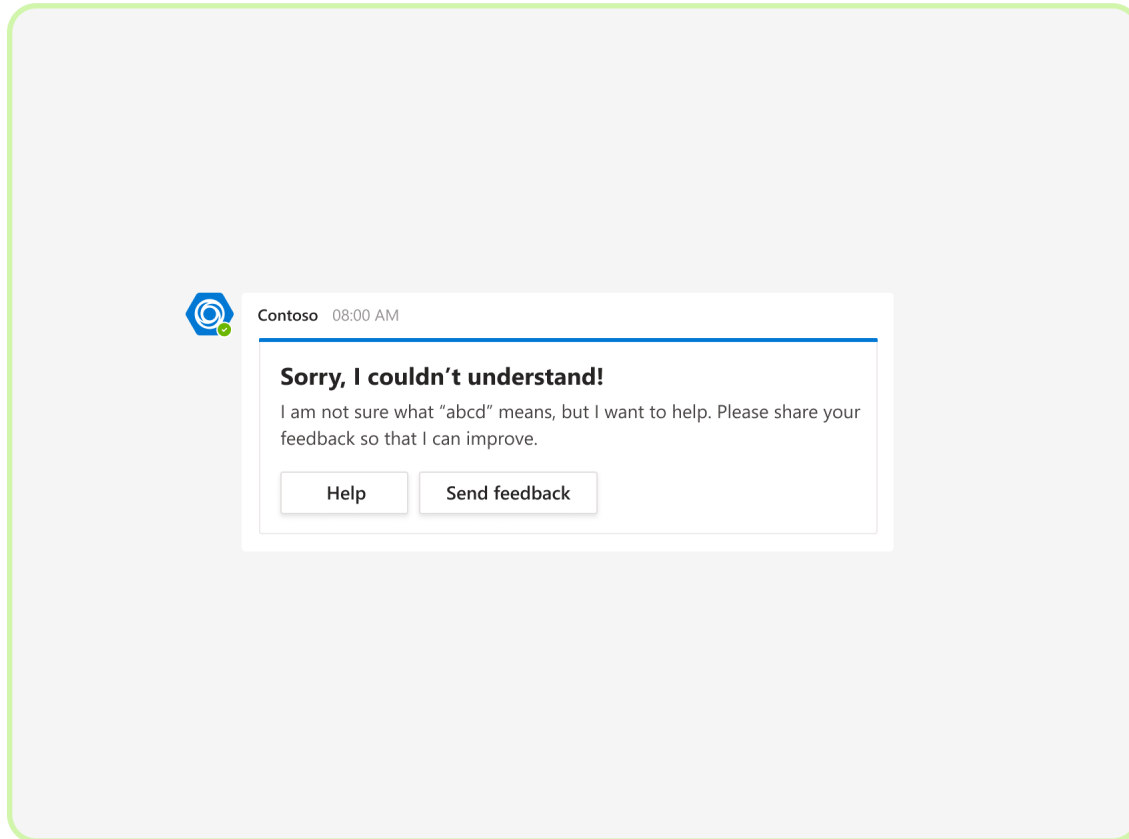
✘ Don't: Use complex textual command formats

Always avoid textual commands with complex formats. Such commands are difficult to remember and prone to typographical errors.

Tip: 6

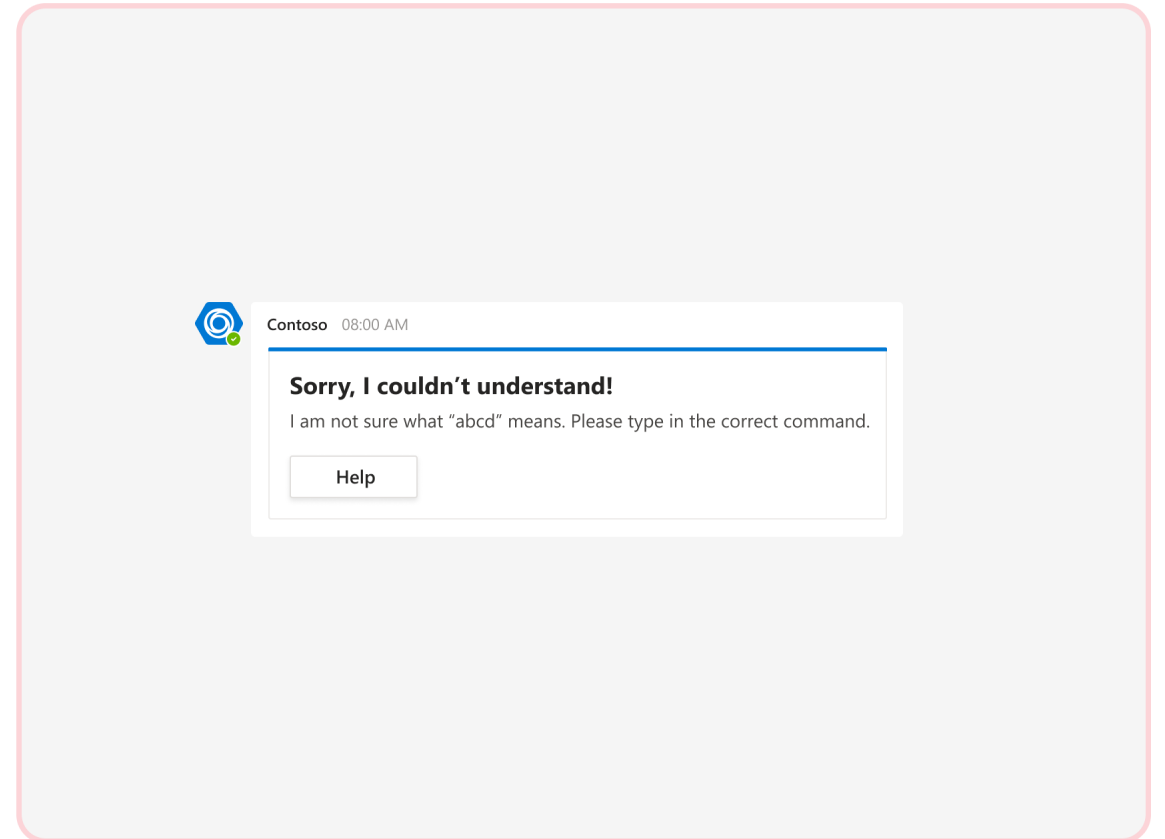
Feedback from users

Feedback



✔ Do: Take feedback from users

In case the bot is unable to understand the commands typed by the user, ask users to share feedback so that suggestions and feedback can be taken directly from the users to improve the next iteration of the app.



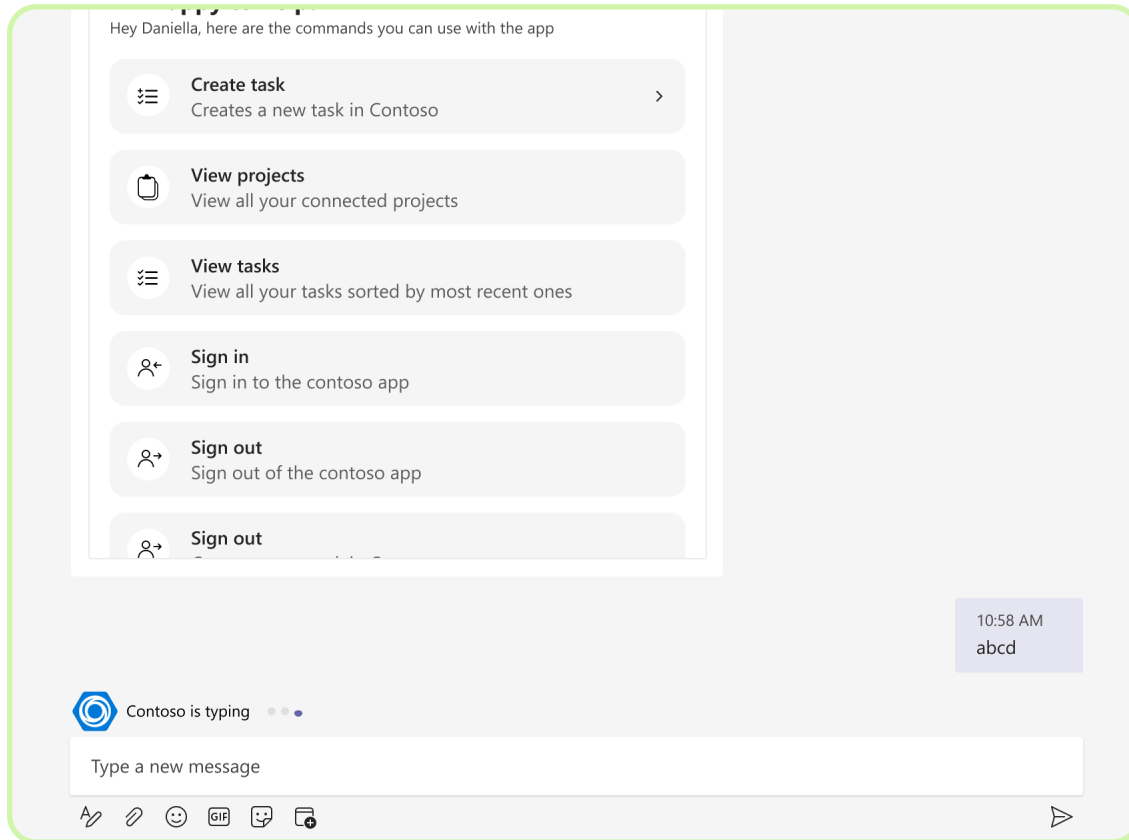
✘ Don't: Miss the opportunities to take feedback

As shown here, it is a good practice to show a relevant CTA in case the bot could not understand the command. However, such events can be further leveraged to seek feedback from the users.

Tip: 7

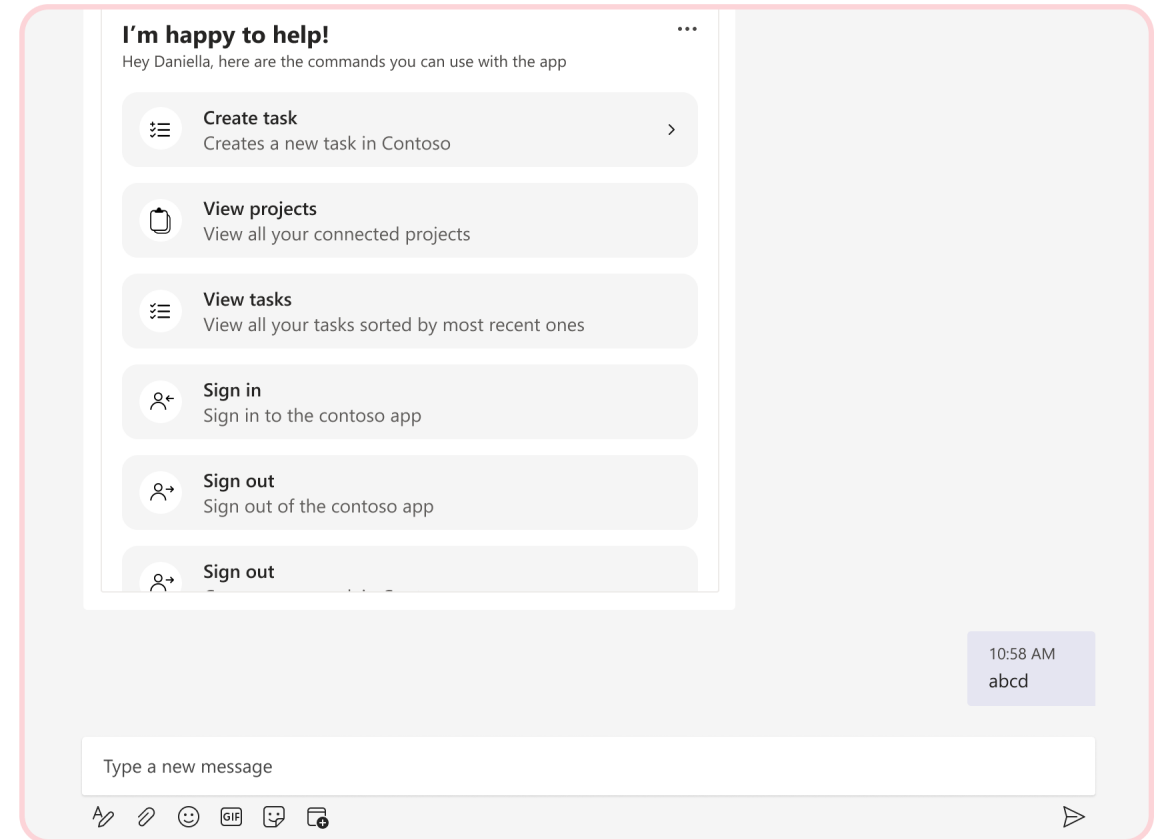
A responsive Bot

Responsive bot



✔ Do: Have a responsive bot

The bot should always respond to any query within 2 seconds. In case the bot takes more time, then a typing indicator should always be indicated.



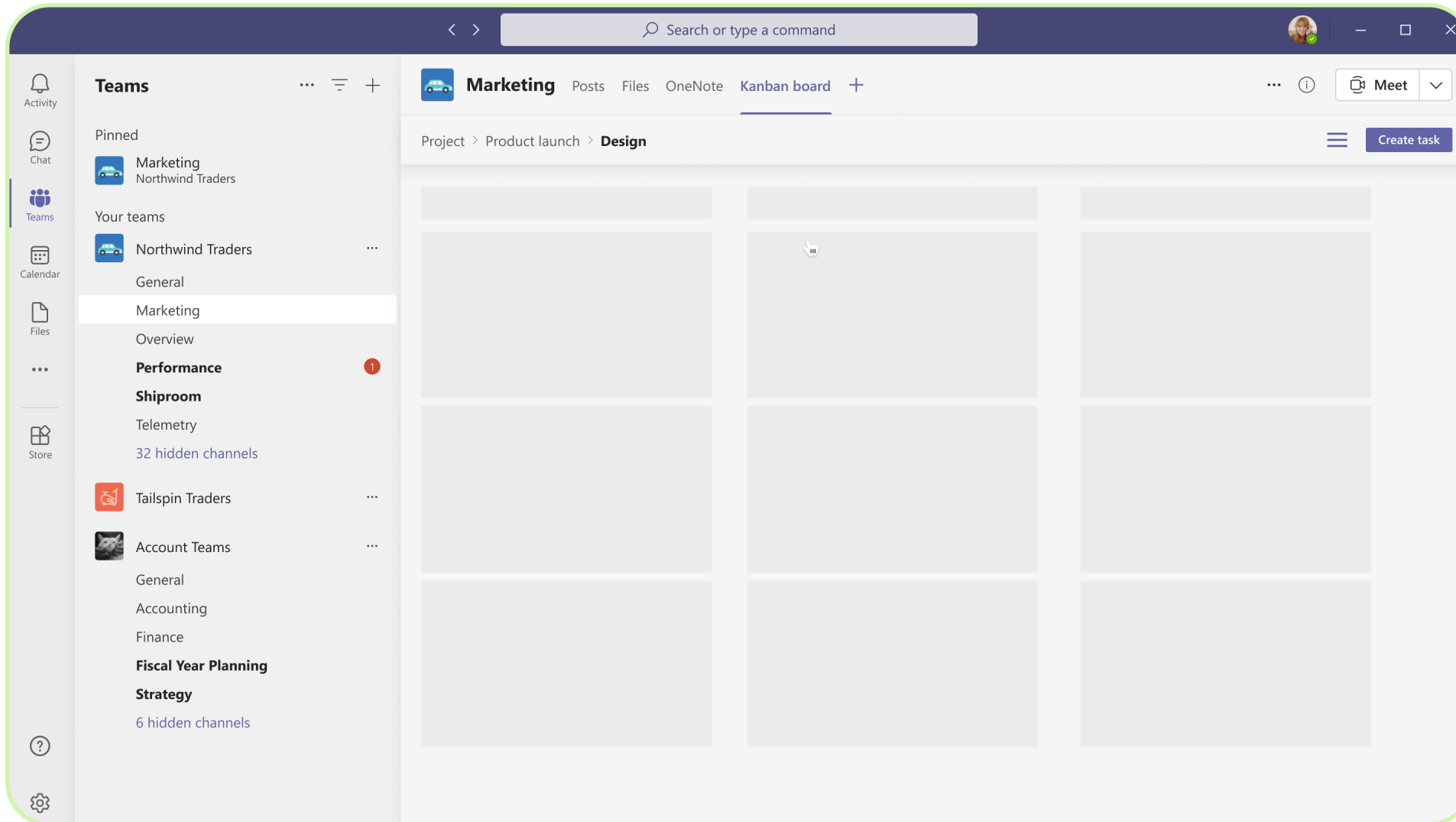
✘ Don't: Keep your users waiting

Avoid showing an empty field from the bot in case there is delayed response from the back-end logic.

Tip: 8

A responsive Tab

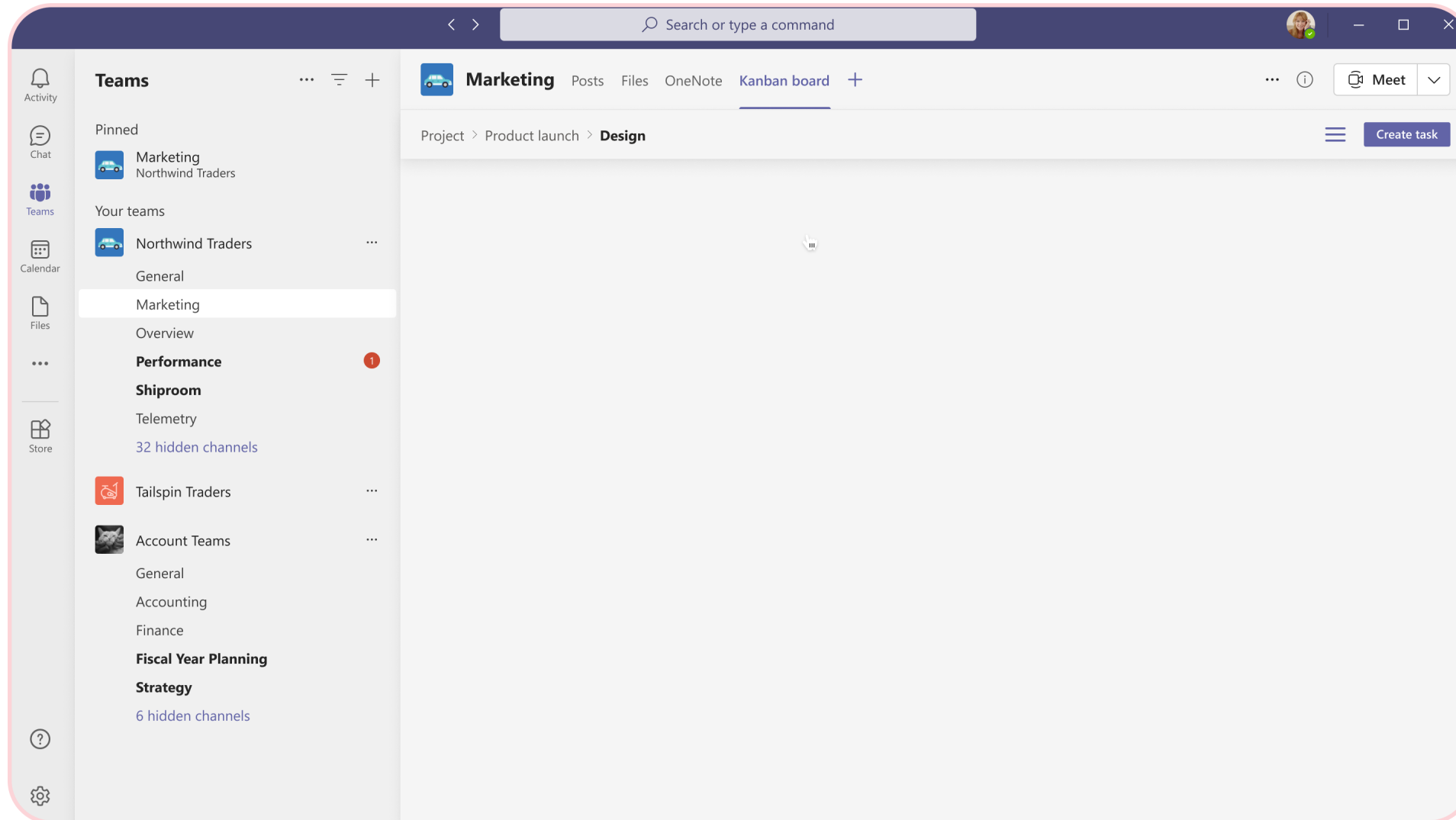
Responsive tab



✓ Do: Have a responsive tab

Any tab should be loaded completely within 5 seconds. In case the tab content takes more time to load, then either show a skeleton of the page or show a loading indicator.

Responsive tab



⊗ Don't: Show empty tab

Avoid showing an empty tab in case there is delayed response from the back-end logic. Always show an active loader or tab skeleton so that the users are assured that there is data coming in.

Tip: 9

Follow design guidelines

Design guidelines

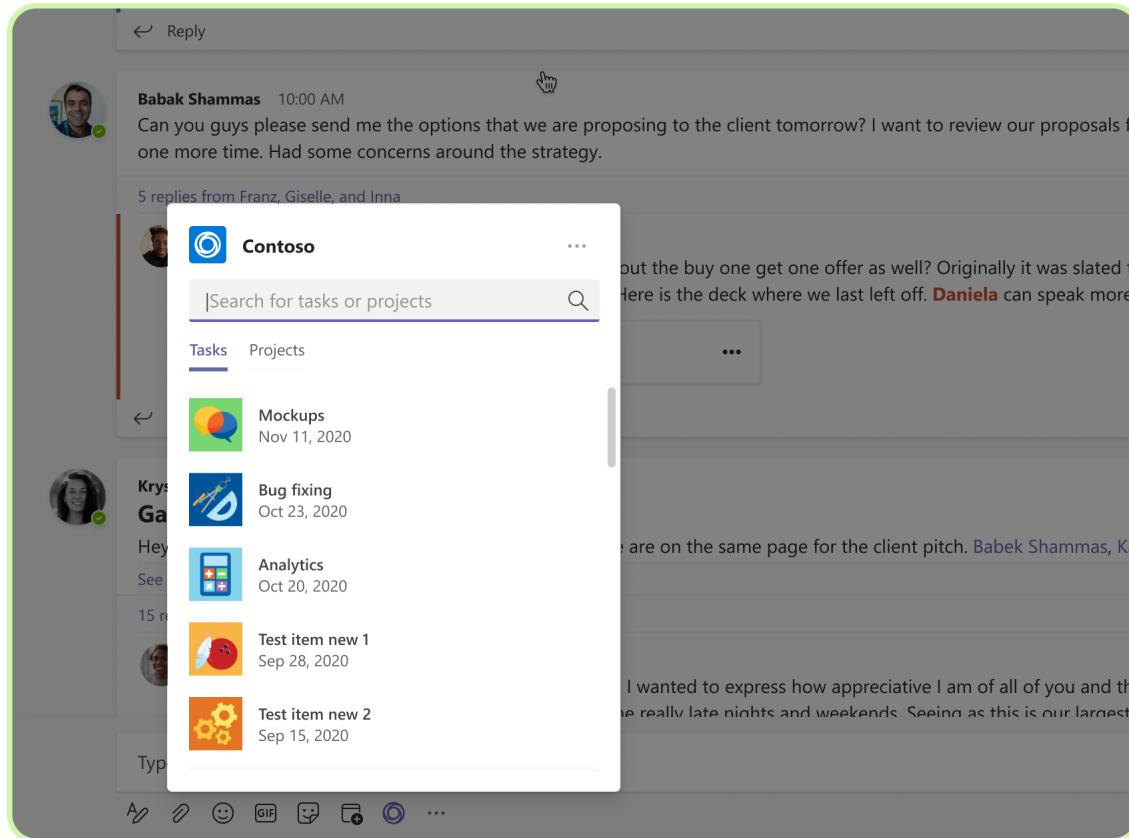
Wherever possible, use the Teams styled components and design guidelines for fonts, colour pallets, grid system, motion, tone of voice etc.



Tip: 10

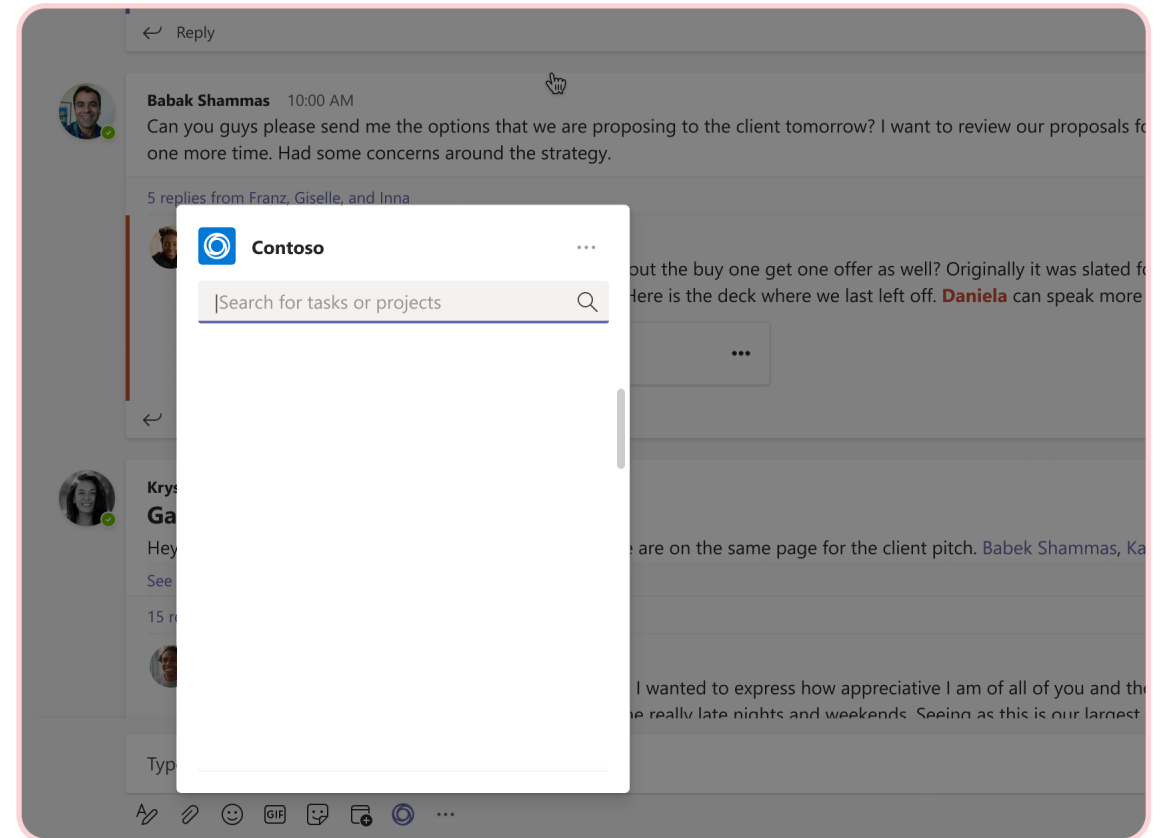
Suggest queries in Message Extension

Message extension



✔ Do: Suggest queries in Message Extension

The search-based message extensions should provide a 0-term query such as "last used" to encourage engagement and provide short cut to the user to the last message extension used.



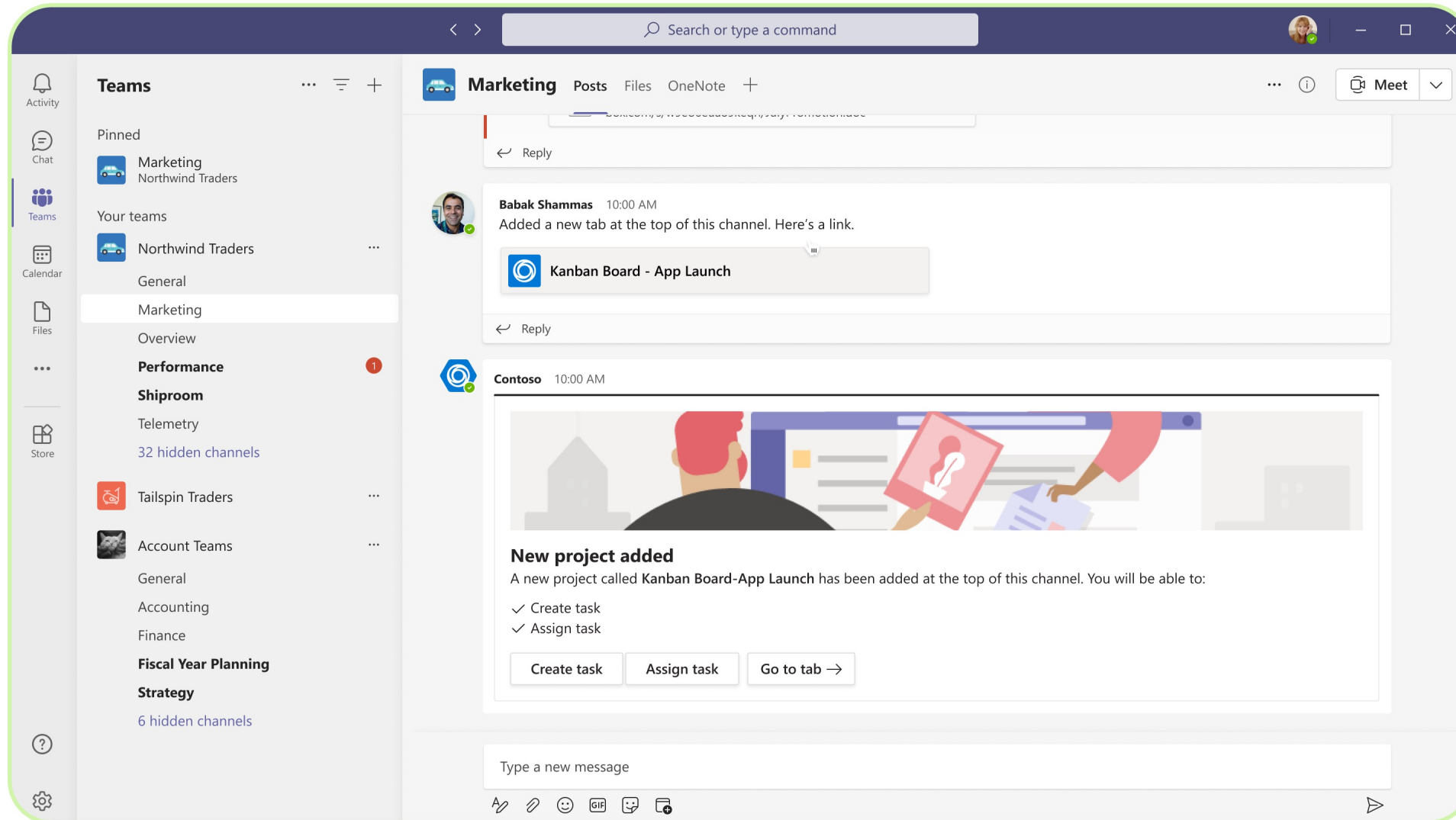
✘ Don't: Show an empty Message Extension screen

It's a bad practice to show an empty area in Message Extension. Prompt suggestions so that users can get themselves acquainted with this capability.

Tip: 11

Bot must provide Tab info

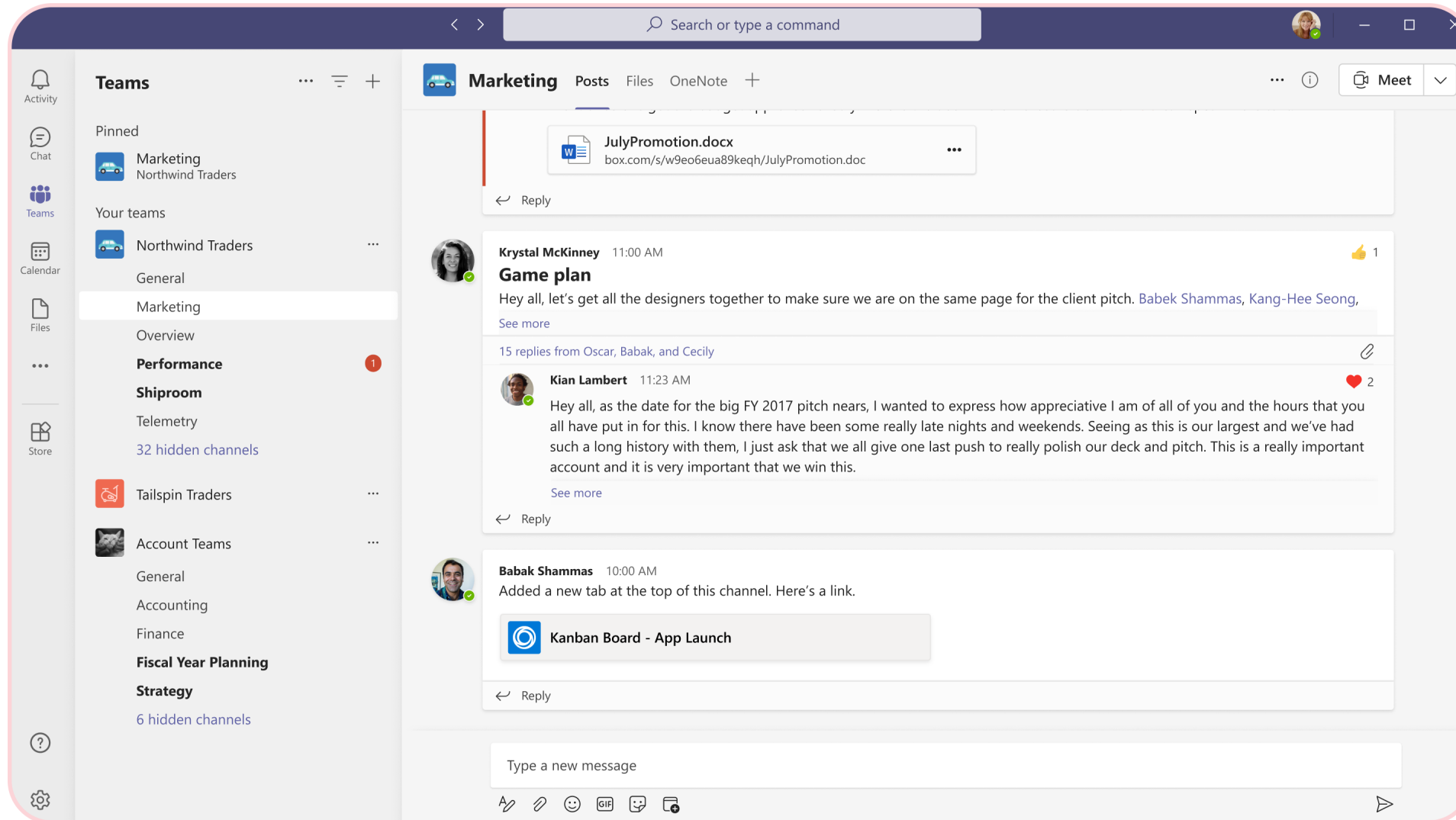
Tab info



✔ Do: Provide pinned tab info via bot

If a tab has been added to a channel that has an app bot associated with it, the bot should provide content that is relevant to the tab that has been pinned.

Tab info



⊗ Don't: Rely on default pinned tab response

Users will never get the context of the pinned tab while reading the default message posted by Teams. Ensure that your bot also posts a welcome card as soon as your app is pinned in collaborative scope.

Thank You

Microsoft Teams Ecosystem

